A Lessons-Learned Knowledge Management System for Engineers

An organizational lessons-learned system facilitates the transfer of knowledge from one project team to another

Stephen Greer
Professional Engineer

Lessons learned is a method of collecting the unique and important knowledge gleaned from experience. While most engineers have heard of this often cited, even cliché term, few really know what this is, and even fewer know how to set up a proper lessons-learned process and use it correctly. The lessons-learned process can be thought of as a smaller planet in the greater universe of corporate knowledge management. In this article, we focus on lessons learned as a way to increase the productivity of the engineering organization, as well as increase the quality of its work product by decreasing costs related to mistakes and rework. But what is it, and how do engineers configure their organization or department to use it efficiently?

Knowledge management systems

Knowledge management systems are being used by an increasing number of organizations, engineering and otherwise, to capture the elusive gains to be realized by mining corporate data and capturing the experiences of its personnel onto computer or other media. Specifically, the lessons-learned process, and more generally, organizational learning, transfers knowledge from the minds of employees and managers into the institution itself to insulate the organization from employee turnover and other losses of experience. However, as applied to engineering and project management, such systems are best utilized by organizations that execute many projects over a long period of time or by engineering firms that do many projects as an integral part of their business operations.

First, we will define some key phrases. A lessons-learned system is a series of procedures culminating in a detailed database that captures unique lessons and knowledge from previous experience, translating it into a format useful for future reference. Knowledge management is a collection of organizational processes that together feature improvements to existing and future operations or dedicated knowledge generated. The end products of these systems are corporate standard operating procedures; construction, design, and operating standards; and function-specific databases of which lessons-learned is one type.

The lessons-learned system

The corporate lessons-learned process in an engineering organization generally requires two things: a useful knowledge-management procedure and a database software to make proper use of the procedure. The database is used to input characteristics and data from all projects with specific data useful to the type, size, client, and most importantly, the useful knowledge gleaned during all phases of design, project management and construction, as well as commissioning and turnover.

It is common that after a large and even successful project a project manager collects the entire team for one final review — a discussion of what lessons were learned, and what to remember for future, similar projects. Usually, the product of such a meeting is a memo or report detailing the mistakes experienced during the project that can be avoided, in addition to summaries of strategies that worked. The memo is then distributed to team members and management. However well-written and thorough it may be, it is ultimately set aside, discarded, or otherwise filed away.

While a good report will effectively capture all the knowledge that should be utilized in future projects, the memo format is an extremely unproductive way to facilitate transfer of that knowledge. Once the summary memo is read and assimilated, with the passage of time, it is likely that the report will be lost or forgotten, and that knowledge will then be subject to the unreliable future recollection of employees. This is true even if the report is kept electronically in a dedicated, though obscure, location on a corporate server. Even in this undesirable scenario, when a new and similar project arises, if the previous team members are not assigned to the new project, then there is very little chance that the new project team can benefit from previous lessons learned.

The core problem in the above scenario is that the knowledge gained from the previous project was not institutionalized. Mistakes made previously are at risk of being repeated on a project if they cannot be easily retrieved and recalled. To avoid these problems, there should indeed be a high-quality report generated at the end of a project detailing the lessons learned, and it should be distributed to all affected employees. However, to utilize lessons learned most effectively, there must also be a system for institutionalizing that knowledge for future use.

A lessons-learned system is a process of recording experience for use at

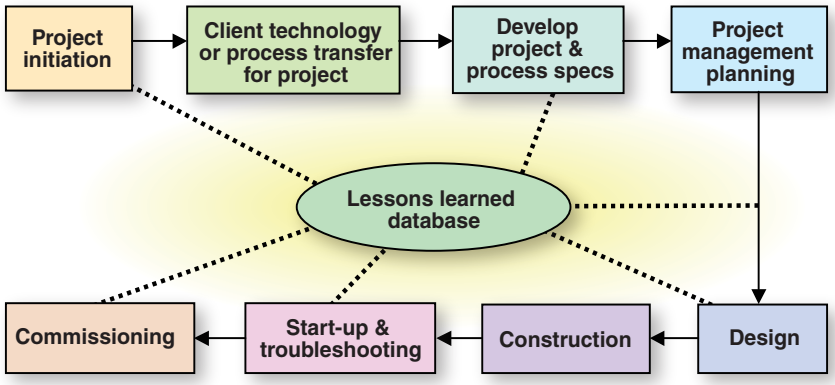

FIGURE 1. An organizational flowchart will help project managers to plan when and how to use a lessons-learned database

a later time when needed. Two things are needed to make it a successful reality: a) an easily searchable repository of information or database, and b) modified or new organizational procedures to make proper use of the database. Though neither step is particularly easy to do and will require an investment of time and money, these investments will have a definite and valuable financial return.

Procedures and organizational schematic

In implementing a lessons-learned system, some background organizational preparation is understood — a culture must be created that rewards reviewing and documenting all things that went wrong or could have been done better, as well as things that went right. This documentation should be done by the project manager or the principle engineer of a project and presented in a professional, detailed, comprehensive report. It takes capable, persistent management at higher levels of the organization to make this happen for every project.

The next step is to consider the organizational changes that are needed to affect the way lessons-learned raw data are used. A standalone database without a structure in place for using information is merely another form of paper memo; it likely won't be used when the situation calls for it, or it will go unused altogether if employees do not understand its proper use. A procedure must be in place to have employees utilize the lessons learned-system at the beginning of a project.

As an example, when an engineer first starts a large project, he or she usually receives a directive from a manager to start doing investigative work, create project goals, develop specifications and initiate client technology-sharing agreements before any design work is initiated. It would make sense that knowledge from previous projects that the engineering firm has done would be most useful at the beginning of a new, similar project. This allows all useful information regarding changes to be folded into the project management when the plans and schedules are being developed.

With this in mind, an organizational flowchart, shown in Figure 1, illustrates when to use such a database. Note that not all organizational steps require, or even recommend, the use of a lessons-learned process. Construction, for example, is often not a core competency of an engineering organization, and this will likely be contracted out to construction firms that have established methods of institutionalizing project lessons. Or, if the firm is vertically integrated, the construction-management division will take over this part.

With a lessons-learned system, the engineering work process might function in the following way: Once a project is initiated, the project manager should, as one of his first tasks, review the database to determine what projects have been done in the past that could be learned from. This could involve anything from the type of project and how it was managed, to specific technologies or designs that will be used on the project and had been used in previous projects. Examples could include piping material compatibilities, a replacement tank project matching the application from several years back, or a specific type of control logic used in a process control system.

Once the project initiation takes place, then subsequent steps, such as the development of project specifications, can be executed in tandem using the lessons-learned system just as any database should be used — for specific information as it is required.

A lessons-learned database

With a new procedure for use in place, the database itself and the information it contains should be understood.

COMPREHENSIVE DATABASE FIELDS

Characterization and identifier fields help project managers efficiently label projects for future searching

- Characterization/Identifier Fields
- Project Name & Number
- Project Type (Utility, Capital/Expense)
- Primary or Secondary Plant or Product Systems or Subsystems
- Raw Materials Involved
- Intermediate Products Involved
- Final or End Products Involved
- Primary Capital Equipment Impacted (Equipment Numbers)
- Secondary Capital Equipment Impacted (Instrument/Control Valve Numbers)
- Material Compatibilities

The availability of several lessons-learned fields encourages extensive documentation of all lessons learned

- Lessons-Learned Fields
- Primary Process-Lessons Learned
- Mechanical Lessons Learned
- Electrical, Instrument & Controls Lessons Learned
- Project Management Lessons Learned
- Structural Engineering Lessons Learned

Today's database software offers robust and cost-effective means for creating a tool for facilitating knowledge transfer. With the use of a computer, it is accessible to all employees, can be simple to use, and can be very effective. Its level of effectiveness is dependent on how it is configured.

In a large organization that executes hundreds of projects over the course of several years, the database must be easily searchable. Projects should be appropriately labeled and characterized using data that will most effectively help locate a project that contains specific information that current personnel would benefit from. Projects must be characterized with as much detail as possible to make the database easily searchable.

Before generating a system for characterizing projects, questions should be raised as to how a specific project can be broken down and labeled:

- What was the project scope and what units were affected: Manufacturing, R&D?
- How much did it cost?
- Was it a capital or expense project?
- What equipment was purchased and installed?
- What was the type and configuration of the controls?
- What was the application?
- Were unique technologies used?
- Were client or corporate processes impacted?

These data are used as variables to allow an exact search to better match an engineer's needs in the future, possibly years later when those who actually played an integral part of a project are no longer there to answer questions.

Depending upon whether such a system is being configured for a product or process-oriented company or an engineering firm, the data identifiers in the box on p. 53 may be used. The characterization fields are variables that allow identification of specific projects in an extensive database. The lessons-learned fields are filled with applicable information. There should also be a separate field to inset a word-processor file for the final report containing a narrative of the lessons learned during the project. A comprehensive database will contain pull-down menus for each variable.

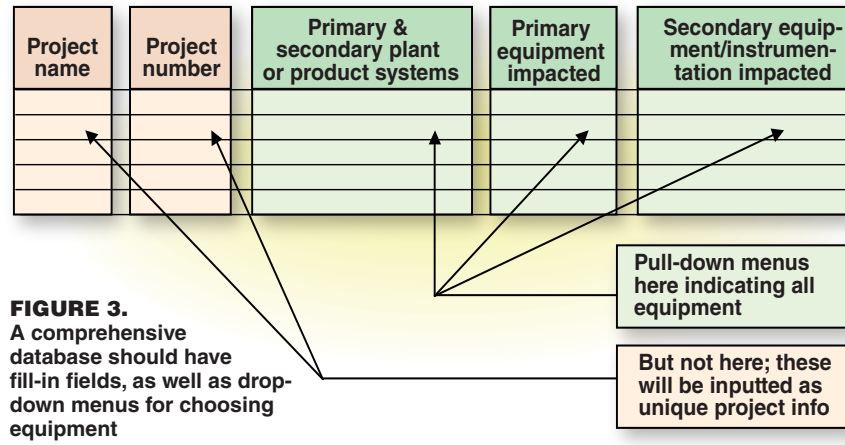
Project name
Project number
Primary & secondary plant or product systems
Primary equipment impacted
Secondary equipment/instrumentation impacted
Pull-down menus here indicating all equipment
But not here; these will be inputted as unique project info

FIGURE 3. A comprehensive database should have fill-in fields, as well as drop-down menus for choosing equipment

For example, under "Primary equipment impacted," there could be several choices such as: 1) main reactor vessel; 2) transfer pump(s); 3) reactor vessel agitator; and so on. For a large plant with thousands of individual pieces of equipment, this could be a daunting task.

One answer to this problem is to link the characterization to a company's computerized maintenance management system (CMMS; for more, see *CE*, Jan., pp. 38–41). Many companies utilize such systems for boosting maintenance unit productivity. Alternatively, the original equipment list can be used to develop the CMMS if the maintenance software does not allow direct access to the entire list. These systems usually have a complete list of a company's hard assets that require maintenance work. The example in Figure 3 illustrates the use of these different headers in a database table.

The most important fields in the database are reserved for the lessons of the project, which should be populated using narrative or shorthand that company employees will recognize. The more advanced database software available today, as well as some versions of the Microsoft Office suite, also allow one to embed a word-processor document into the same entry. In other words, the same valuable project lessons-learned report described earlier in this article can be inserted into the database project entry and thus is accessible for as long as the database is active. The memo can simply be called up when the project file is located and the rest of the raw data is reviewed.

Of course, a database is useless if it is not properly updated. A firm's management must ensure that, at the end of a project, the project team develops a report, and that the project manager inputs project findings into the database. There is also the issue of network load resulting from the database, especially if it is large and requires substantial costly memory. Ownership of the database should therefore lie within an organizational unit that normally performs database management such as a maintenance unit, which may require a dedicated employee to affect this.

A lessons-learned procedure, coupled with today's highly capable databases, has the potential to measurably increase the productivity of a project-intensive organization and the quality of the services they produce; to not pursue one is to deny your organization its valuable benefits. ■

Edited by Kate Torzewski

References

1. Construction Industry Institute, University of Texas at Austin, "Modeling the Lessons Learned Process."

Author

Stephen Greer (sag11@att.net) is a professional engineer licensed in the state of Michigan where he currently resides. He holds a B.S. from Michigan Technological University in Houghton, Mich. from which he graduated cum laude. He has worked in the specialty chemicals; engineering services; and pharmaceutical and biopharmaceuticals industries where he specialized in developing and executing capital projects in process optimization; facility expansion; utility systems; and pump networks and systems as a project and process engineer. He also takes an interest in the quality and efficiency of delivering engineering services within the corporate environment.